OPTIMIZATION APPROACHES TO THE PROBLEM OF EDGE LINKING WITH A FOCUS ON PARALLEL PROCESSING*

M. D. Diamond, N. Narasimhamurthi, and S. Ganapathy

Department of Electrical and Computer Engineering
University of Michigan
Ann Arbor, MI 48109

ABSTRACT

An important problem in computer vision, that of edge linking for contour or line drawing extraction, is approached from the point of view of a graph labeling problem. A Lagrange dual approach to an integer programming formulation of this problem will be presented. Although the inherent complexity of the problem will not be reduced, the techniques presented below will allow for a partial decomposition of the solution algorithm. Furthermore, the approach which will be presented appears to have certain advantages over existing line tracking and graph searching algorithms.

I INTRODUCTION

Two basic ideas will be pursued in the following discussion: The first involves an approach to the problem of edge linking. The second involves an introduction to the use of dual methods for combinatorial optimization problems relevant to issues of artificial intelligence. In order to relate these two concepts, edge linking will be modeled in terms of a graph labeling problem which will then allow for the direct application of the proposed dual methods.

A. The Problem of Edge Linking in Computer Vision

The derivation of contours describing the outlines and the salient features of objects Is an essential component of most model based systems. Once these outlines have been derived, well established classification techniques, such as syntactic pattern recognition based spline primitives (Perkins, 1979, Fu 1974), Fourier shape descriptors (Zahn and Roskles, 1972), or generalized Hough transforms (Ballard, 1981), can be used to recognize the object from a set of known models. It is well known, however, that the edge detection process Is not in Itself sufficient to describe the required contours, as a primitive edge map will almost always contain broken contours or spurious edges. It is for this reason that edge linking is considered to be one of the most basic problems in computer vision (Marr, 1975).

Unfortunately, the problem appears to be as difficult as It is important. Although, the inherent difficulty does not appear to be so much in the conceptual problem of finding a good model to represent the contour or signal, in the presence of noise, but rather with the computational complexity of the algorithms which extract it. For example, in one model based system, the edge extraction and linking required approximately 99% percent of the overall computational effort (Perkins, 1979). In fact, most current edge linking approaches are based on combinatorial search or tracking algorithms wherein the points in the state space correspond to pixels in the image (Ballard and Brown, 1982). The search is usually guided by an evaluation function which is in some manner related to the over all length or strength of the contour in the original edge map.

It is not surprising that computational difficulties arise in conjunction with a search defined on a space the size of a typical image. If it is assumed, therefore, that edge linking is inherently a problem in combinatorial optimization, and on the other hand, a fundamental problem in computer vision, then research into means by which the search can be implemented in a highly parallel manner, despite the implications for the cost of the supporting hardware, is justified. Another basic problem with existing strategies Is that they may encounter difficulties if, for example, the search is Incorrectly rooted, as will occur if the starting point happens to be a noise point, or if, because of some ancillary feature in the image, the search continues down a strong, but false path. The focus of the work described below is in addressing these basic issues.

B. Dual Approaches to Combinatorial Search Problems

Because it does not appear that existing graph searching methods can be implemented in a parallel manner, the approach proposed here is to perform the search in an associated dual space. In order to do so, the problem is first formulated as an integer programming problem. A Lagrange dual (Bazaraa and Shetty, 1 979) realization of the problem is then presented. In accordance with price directed decomposition theory (Shapiro, 1979), the particular form used will allow for the decentralized computation of the solution, at least in the initial stages of the process. Although graph searching problems occur In almost every area of artificial intelligence, the techniques presented here are particularly suited to those combinatorial search problems where the entire state space is known in advance. A graph labeling problem, where the graph is defined with respect to an Image raster, is an example of this situation.

[X] This work was supported In part by the Robotics Research Laboratory, and In part by the Ultra sonics Imaging Laboratory, both In the Department of Electrical and Computer Engineering, University of Michigan

[xx] Current address: Computer Systems Research Laboratory, Bell Telephone Laboratories, Holmdel, New Jersey

C. Graph labeling and the edge linking problem

A graph labeling problem is one in which a unique label λ_j from some set, Λ, of labels must be assigned to each vertex of a graph, $G = (V,E)$, with vertex set $v = \{v_1, v_2, \cdots, v_n\}$, and an edge set $E \subseteq V \times V$. In the application example of section IV, the graph is defined in terms of an image with each pixel considered to be adjacent to its eight immediate neighbors. The label set in this case corresponds to the assertion of the existence of scene events, which may include line segments, elbows, and comers at various orientations at each pixel, as shown in figure 1. A constraint network (Montanari, 1974, Mackworth, 1977, Freuder, 1978) is constructed from the graph G by associating with each edge $v_i v_{i'} \in E$ a binary relation, $R_{ii'}$, known as a *constraint relation*, defined on the label set, $R_{ij} \subseteq \Lambda \times \Lambda$. A pair of labels (λ_j,λ_j') on adjacent vertices v_i and $v_{i'}$ are *consistent* if $(\lambda_j,\lambda_j') \in \Lambda$, and *inconsistent* otherwise. For this application, a pair of labels is consistent if and only if an edge segments is not broken across a pixel boundary as given, for example, in Diamond et al. (1982). A *labeling* $\bar{\lambda} = \lambda^1 \lambda^2 \cdots \lambda^n$ which assigns label λ^i to vertex v_i is consistent if every pair of labels on adjacent vertices is consistent. Note that within the context of this application, a labeling will be consistent if and only if it describes a smooth contour. Associated with each label λ_j on each vertex v_i is an initial labeling value, or merit figure, c_{ij}. For this application, the merit figures are assumed generated by feature detectors sensitive to the various scene events as shown in figure 1.

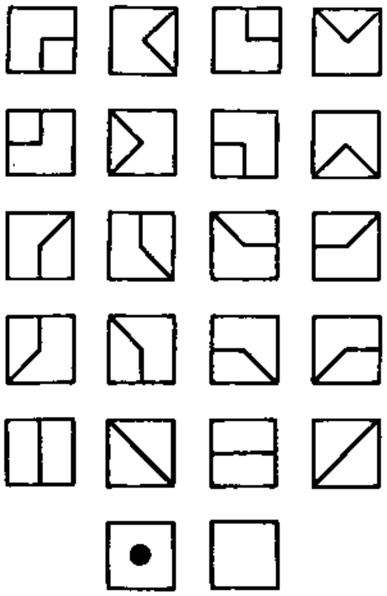


Figure 1: primitive features corresponding to labels for the edge linking application.

Although several viewpoints on the continuous graph labeling problem have been presented (Rosenfeld et al. 1976, Ullman 1979, and Davis and Rosenfeld 1976), In the definition used here, the problem will be to find that consistent labeling such that the sum of the initial labeling value Is maximum. This definition of the problem is attractive because a solution can be extended to many established classification rules (Diamond et al., 1982). Finally, in the same spirit as Ullmans "simple local processes," (Ullman, 1979) we will be interested in developing decentralized algorithms which can be Implemented on an SIMD or "cellular" architecture.

As defined above, continuous graph labeling is easily shown to be NP-complete. The combinatorial nature of the problem will preclude, therefore, the derivation of a local or cooperative algorithm which guarantees a globally consistent labeling in the general case. This, then is the major difficulty of the edge linking strategy proposed here. However, as will be demonstrated, the technique described below can be used to significantly improve an initial, noisy edge map, and if a globally consistent labeling Is required, the dual approaches suggested below may be used as a front end for further processing. In this sense, these techniques may be thought of in the same vein as the discrete relaxation processes (Rosenfeld et al., 1976, Waltz, 1975) extended to the continuous domain, which serve to reduce the combinatorial search space but do not necessarily guarantee a globally consistent labeling.

II A Lagrange Dual Approach to the Graph Labeling Problem

A. General Formulation

For the purpose of the following discussion, it is assumed that the graph labeling problem is defined with respect to a graph with n vertices, $|V| = n$, and a label set with m labels, $|\Lambda| = m$. Indices such as j and j' will always be used in conjunction with labels, and λ_j and $\lambda_{j'}$ will denote elements of the label set Λ. In a similar manner, indices such as i and i' will always be used in conjunction with vertices, and v_i and $v_{i'}$ will denote elements of the vertex set V. Let $c_{ij} \in R$ be the initial strength measure or merit figure associated with label λ_j on vertex v_i. The integer programming formulation of the continuous graph labeling problem is then given by:

Maximize:

$$\sum_{v_i \in V} \sum_{\lambda_j \in \Lambda} c_{ij} \, x_{ij} \qquad 2.1$$

Subject to:

$$\sum_{\lambda_j \in \Lambda} x_{ij} = 1, \quad i = 1,\ldots,n \qquad 2.2$$

$$x_{ij} + x_{i'j'} \leq 1 \qquad 2.3$$

for every inconsistent pair of labels $(\lambda_j,\lambda_{j'})$ on adjacent vertices v_i and $v_{i'}$.

$$x_{ij} \in \{0,1\}, \quad i = 1,\ldots,n, \quad j = 1,\ldots,m \qquad 2.4$$

As given by constraint (2.4), x_{ij} is a zero-one decision variable associated with label λ_j on vertex v_i. In the solution, $x_{ij} = 1$, if label λ_j is to be selected for vertex v_i. Thus, constraint (2.2) serves to guarantee that exactly one label is assigned to each vertex and constraint (2.3) serves to specify that for adjacent vertices v_i and $v_{i'}$ a pair of labels λ_j and $\lambda_{j'}$ are not simultaneously assigned to these vertices if they are not consistent.

Let

$$x = (x_{11}, x_{12}, \ldots, x_{nm})$$

be the vector of decision variables. Furthermore, let μ be the set of all four-tuples $iji'j'$ corresponding to invalid pairs of labels $(\lambda_j, \lambda_{j'})$ on adjacent vertices v_i and $v_{i'}$. The dual will be defined with respect to a vector u, which consists of components $u_{iji'j'}$, *one such component for each* $iji'j' \in \mu$. According to the theory, the primal vector x and dual vector u are used in the definition of an auxiliary function:

$$\varphi(x,u) = \sum_i \sum_j c_{ij}\, x_{ij} - \sum_{iji'j' \in \mu} u_{iji'j'} (x_{ij} + x_{i'j'} - 1) \qquad 2.5$$

By simple algebraic manipulation, $\varphi(x,u)$ is seen to have the form

$$\varphi(x,u) = \sum_{v_i \in V} \sum_{\lambda_j \in \Lambda} r_{ij}\, x_{ij} + \sum_{iji'j' \in \mu} u_{iji'j'} \qquad 2.6$$

where

$$r_{ij} = c_{ij} - \sum_{iji'j' \in \mu(i,j)} u_{iji'j'} \qquad 2.7$$

is the current merit figure or labeling value, *relative* to the dual vector u. Also, $\mu(i,j) \subseteq \mu$ is the set of all 4-tuples corresponding to invalid pairs of labels, in which the label λ_j on vertex v_i participates. The Lagrange dual form of the original problem is then defined to be:

$$\text{Minimize: } \Theta(u), \quad u \geq 0$$

Where:

$$\Theta(u) = \sup_{x \in X} \{ \varphi(x,u) \} \qquad 2.8$$

$$= \sup_{x \in X} \{ \sum_{ij} r_{ij}\, x_{ij} + \sum_{iji'j' \in \mu} u_{iji'j'} \}$$

Here, X is the set of all 0-1 vectors x, which correspond to (unambiguous) labelings, that is, constraint (2.2) is satisfied by every $x \in X$. *Note, however, that* $x \in X$ *need not be a consistent* labeling. Thus x need not necessarily satisfy constraints (2.3).

The form of the dual given by equations (2.8) allows for local determination of those $x \in X$ at which the maximum value of $\varphi(x,u)$ occurs, for a given u. This is because the r_{ij} can be computed on a local basis, and $\sum u_{iji'j'}$ is independent of x. Thus deriving the *primal* values associated with a given u is equivalent to choosing a label, λ_j at each vertex such that the current labeling value r_{ij} is maximal, a process often referred to as local maxima selection (Zucker et al., 1978).

B. Example Problem

A simple example to illustrate the formulation and notation used above is *given* in figures 2a, and 2b. Figure 2a is a $|V| \times |\Lambda|$ product graph gives each label on each vertex explicitly, as well as the pairs of consistent labels by connecting them with a solid line. Figure 2b shows the complement of this graph, in which *inconsistent* pairs of labels are connected by dotted lines. Because of the definition of the dual function, it becomes more important to concentrate on the complement graph.

The primal statement of this problem becomes:

Maximize:

$$c_{11}x_{11} + c_{12}x_{12} + c_{13}x_{13} + c_{21}x_{21} + c_{22}x_{22} + c_{23}x_{23}$$

Subject to:

$$x_{11} + x_{12} + x_{13} = 1 \qquad 2.9$$

$$x_{21} + x_{22} + x_{23} = 1$$

$$x_{11} + x_{21} \leq 1$$

$$x_{12} + x_{21} \leq 1 \qquad 2.10$$

$$x_{11} + x_{22} \leq 1$$

$$x_{ij} \in \{0,1\}. \qquad 2.11$$

The Lagrange dual formulation is:

Minimize:

$$\Theta(u_{1121}, u_{1221}, u_{1122}) = \max_{x \in X,\ u \geq 0} \{ \varphi(x,u) \},$$

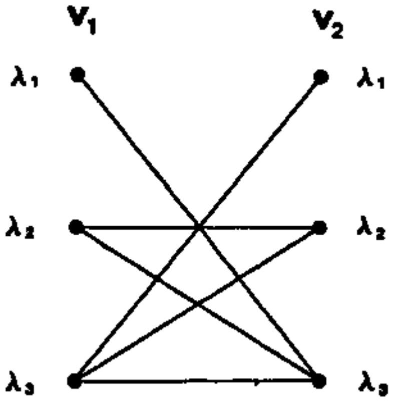


Figure 2a product graph for the constraint network of the example.

with:

$$\varphi(x,u) =$$

$$x_{11}(c_{11} - u_{1121} - u_{1122}) + x_{12}(c_{12} - u_{1221}) + x_{13}c_{13} +$$

$$x_{21}(c_{21} - u_{1121} - u_{1221}) + x_{22}(c_{22} - u_{1122}) + x_{23}c_{23} +$$

$$u_{1121} + u_{1122} + u_{1221}$$

In view of equation (2.10), the definition of $\varphi(x,u)$, and figure 2b it can be seen that there is a dual variable associated with each such edge between invalid pairs of labels. The current labeling value associated with each label on each vertex is the the initial labeling value minus the sum of all dual values associated with edges in the dual graph.

C. PMI Sets

The continuous graph labeling problem can be reformulated by combining constraints, to the extent possible on a local basis. For example equation (2.10) of the example in section 2.2, can be replaced by

$$x_{11} + x_{12} + x_{21} \le 1 \qquad 2.10a$$

$$x_{11} + x_{21} + x_{22} \le 1$$

Note that the set consisting of labels λ_1 and λ_2 on vertex v_1 and label λ_1 on vertex v_2 has the property that only one of these labels can be chosen at once. The same holds for the set consisting of label λ_1 on vertex vi and labels λ_1 and λ_2 on vertex v_2. Such constraints will be referred to as *pairwise maximally Inconsistent* (PMI) sets of labels. The PMI sets can be determined, in general, by considering the product graph with the addition of edges connecting every label on a given vertex with every other label on that vertex. The PMI sets then correspond to the maximal cliques of this graph (refer to figure 3) .

In the problem formulated in terms of PMI set constraints, there is a dual variable associated with each PMI set. In general there will be fewer PMI sets than pairs of invalid labels between a given pair of vertices. Therefore, by restructuring the problem in this manner, there will also be fewer dual variables, and this will reduce the amount of memory required for a given application. An advantage in computation can also be gained as will be discussed below. Finally, with this formulation, a relationship exists between this problem and the discrete relaxation processes, which can be used to understand the behavior of the solution algorithms.

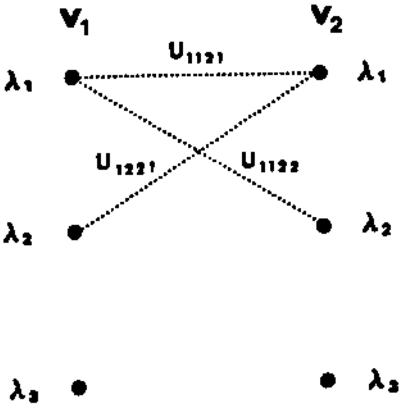


Figure 2b: complement graph for the constraint network of the example.

III MINIMIZING THE DUAL

We present below an outline of the algorithm used to minimize the dual function. Further details are available In a preliminary report (Diamond, 1983), although work on certain aspects of this approach is still in progress.

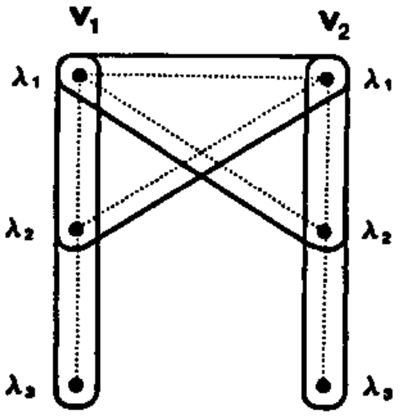


Figure 3 : augmented complement graph showing maximal cliques. Edges exist between label λ_1 and λ_3 on vertex v1 and λ_1 and λ_3 on vertex v2 but are not shown.

[3] Note that there will always be a maximal clique associated with the set of all labels at a given vertex. The resulting constraint $\sum_{\lambda_j \in \Lambda} x_{ij} \le 1$ is redundant, in view of equations (2.2).

A. The Descent Algorithm

Each iteration of the descent procedure involves one "pass" over the image. A pass involves considering all possible pairs of adjacent vertices, or equivalently, edges in the graph. Any set of disjoint pairs of vertices (or edges) can be considered simultaneously, hence the partial decomposition of the algorithm. In considering any such pair, say vertex v1 and vertex v1 the following procedure is used:

[1] Calculate the current labeling values, rij, for all labels on vertex V1 and V1'.

[2] Let mi be the maximum of all current labeling values associated with labels on vertex V1. Likewise, let m, be the maximum of all current labeling values associated with labels on vertex vi'.

[3] Define the set M; to be the set of labels on vertex v, whose current labeling value is equal to nrt|. Likewise, define the set M| to be the set of labels on vertex vj whose current labeling value is equal to my.

[4] If the sets of labels, M$_t$ and Mj are not contained in a PMI set, then no local descent direction exists, otherwise,

[6] Let L| be the set of labels in a covering PMI set associated with vertex Vj and Lj be the set of labels in that set associated with vertex *vi.* Let

$$\bar{L}_i = \Lambda - L_i, \quad \bar{L}_i' = \Lambda - L_i'.$$

[6] Define

$$\bar{m}_i = \max\{c_{ij} : \lambda_j \in \bar{L}_i\}, \quad \bar{m}_i' = \max\{c_{i'j'} : \lambda_j' \in \bar{L}_i'\}.$$

[7] Define

$$\delta_i = m_i - \bar{m}_i, \quad \delta_i' = m_i' - \bar{m}_i'$$

and finally,

$$\delta = \min\{\delta_i, \delta_i'\}$$

[8] The descent step is implemented by adding *6* to the dual variable associated with the covering PMI set constraint.

An interpretation of this algorithm can be given as follows: Initially, one chooses that label at each vertex with the greatest associated labeling value, that is, according to the local maxima selection process. If the resulting labeling is consistent then the original problem has been solved, that Is, the resulting labeling is that globally consistent labeling such the sum of the Initial labeling values is maximum. Otherwise, the strategy involved is to "penalize" the labeling values associated with Invalid pairs of labels across adjacent vertices in the graph. This penalty Is implemented through equations (2.7) by Increasing the value of the associated dual variable. The hope Is, that reducing the values of the labels which participate in inconsistent labeling pairs will, at the next Iteration, result In different labels being chosen at the respective vertices. This process will then reduce the overall number of Inconsistent labeling pairs.

Although the algorithm appears on the surface to be quite detailed, In fact It is very easy and efficient to Implement. By bit encoding the labels at each vertex participating in a given operation (e. g., sets M1 and M|) as well as the labels in a given PMI set, the processing required for a given pair of vertices can be reduced to a couple dozen machine instructions per vertex pair per iteration. For the label set discussed in the introduction, only 8 PMI sets are required to cover all the constraints between a given pair of vertices, so the search required to find a cover is minimal. Finally, from the basic operations involved it can be estimated that this algorithm will run well over 100 times faster than any of the algorithms associated with the so called "relaxation labeling" processes.

B. Spacer Steps

One of the immediate drawbacks with the algorithm given above is that it will generate fixed points with 2 or more labels with associated labeling values tied for the greatest value at each vertex in the graph. When this happens, and when the resulting sets of chosen labels cannot be covered by a PMI set at any vertex, then the dual can not be minimized any further on a local basis. In this case, it is not possible, in general, to choose a label from among those which are tied with the best current labeling value except on the basis of informal heuristics. For this reason a "spacer step" will be used after every iteration of the descent algorithm given above. In the experiments described below the spacer step consists of an Iteration of the "average-max" updating rule (Diamond, 1983) which updates the current labeling value cj of a label Xj on vertex v, by:

$$c_{ij}^{t+1} = \frac{1}{N+1}\left[\left(\sum_{v_j \in N(I)} \max_{\lambda_k \in \Lambda}\{r_{ij}(\lambda,\lambda')\, c_{jk}^{t}\}\right) + c_{ij}^{t}\right].$$

Where $N(I)$ Is the set of vertices adjacent to vertex v_i, $r_{ii'}(\lambda J,\lambda_{j'})$ is one If label λ_j on vertex v_i Is consistent with label $\lambda_{j'}$ on vertex $v_{i'}$ and zero otherwise, and $N = |N(I)|$.

Thus, a processor performing the updating for label Xj on vertex v, would generate N values to be averaged (along with the current labeling value c_{ij}^{t}), one such value corresponding to each vertex in the neighborhood, by taking the maximum of the current labeling values associated with labels consistent with λ_j on vertex vj.

If $\bar{\lambda}^t$ Is the current labeling chosen by the local maxima selection process and $c(\bar{\lambda}^t)$ is the sum of the associated current labeling values, then the average-max updating rule can be shown to have the following properties:

[1] If it is not a consistent labeling, then the sum of the current labeling values with $\bar{\lambda}^t$ will have decreased after the next iteration, that is $c(\bar{\lambda}^{t+1}) < c(\bar{\lambda}^t)$.

[2] If $\bar{\lambda}^t$ is consistent then $c(\bar{\lambda}^{t+1})$i = $c(X^t)$ and the labeling selected at each iteration t > t will be the same.

[3] From [1] and [2] the value $c(\bar{\lambda}^t)$ is a non-increasing function of t. Furthermore, it can be shown that the value associated with a consistent labeling will increase at each iteration, If that labeling is not the one selected by the local maxima selection process.

IV EXPERIMENTS

Figure 4a shows a scene taken from the General Motors, "Bin of parts" database after the application of feature detectors sensitive to each of the scene events shown in figure 1. Those labels at each pixel such that the corresponding feature detector outputs are maximal are shown in this figure. Figures 4b and 4c are the results of choosing the best labels at each pixel after 8 and 20 iterations of the procedure described above. In the case of labels tied for the best at each vertex, preference was given to the label corresponding to the blank pixel. Otherwise, that label which was consistent with the greatest number of labels on vertices in the neighborhood was chosen. Beyond this, ties are broken arbitrarily.

V DISCUSSION

We have described an approach to the problem of edge linking which is based on the graph labeling model. A Lagrange dual approach to the integer programming formulation of the associated continuous graph labeling problem has been discussed. Obviously much detail has been left out as the intention here has been to limit this presentation to an overview of current results related to an ongoing research effort.

The approach described here, as is the case with many of the other cooperative algorithms applied to problems in computer vision, the relaxation labeling processes in particular, is heuristic in nature. Nonetheless, the existence of an underlying problem definition as well as the direct relationship between the model and the application is a major advantage which this approach has over the relaxation labeling algorithms. In the latter case such a definition is not present. There are other advantages. For example, because the dual function is convex, one can with certain precautions, guarantee that the descent algorithm converges. Although we can offer no theoretical limit on the convergence time (i.e. the number of iterations required before the labeling chosen by a local maxima selection process does not change), it has been observed that the algorithms described here will converge on the order of 16 to 20 iterations for the examples given above. These figures appear to be independent of the particular problem.

However, there *are* also problems in using heuristic techniques to solve a combinatorial optimization problem In a decentralized manner. For example, these algorithms will almost always result in a fixed point with multiple labels tied for the maximum value. As noted previously, in this case there is no basis on which to make an intelligent labeling choice in a purely local manner even though there may exist a globally consistent labeling among the labels tied for the best value. Finally, the process is somewhat sensitive to the amount of noise in the image, and very sensitive to the way in which the initial labeling values are derived.

If the main interest is in the application itself, then a more sensible approach would be to relax the requirement of a totally decentralized solution, that is, to Incorporate some form of an enumeratlve scheme as with the graph searching methods described in the Introduction. Even so, the graph labeling model of the edge linking application offers an advantage since many aspects of a branch and bound approach can then be implemented In a decentralized manner. For example, a variation of the Lagrange dual algorithm presented here could be used as a means for generating bounds in a branch and bound approach (Geoffrlon, 1974), and the discrete relaxation process (Rosenfeld et al., 1976) could be used as part of a feasibility test of a given candidate subproblem.

Furthermore, this model offers the potential of adapting techniques from other classes of 0-1 integer programs to the edge linking application. The graph labeling problem, as defined here, is a special case of the well established *set partitioning* problem (Balas and Padberg, 1974). It is also easily transformed to the vertex and set packing problems, as well as the set covering problem, for which heuristic algorithms to handle large scale situations exist.

In order to make the general model more useful, a better understanding of the graph labeling problem is needed. Current efforts towards a formal understanding of this problem involves both an investigation into the nature of its linear programming relaxation, which is the problem of section 2.1 with equation (2.4) replaced by:

$$x_{ij} \geq 0 \quad i = 1,\dots,n, \; j = 1,\dots,m,$$

and the relationship between the Lagrange dual and this relaxation. Although, aspects of the algorithm presented above are still not understood, the results from the application of these techniques to real world scenes, for example, the industrial scenes shown in section 4, are encouraging. Finally, some effort is being directed towards developing techniques for solving the problem when the underlying graph has a particular regular structure such as that corresponding to an image, which has pronounced rows and columns. When this occurs, a bidirectional dynamic programming approach (Diamond, 1983) can be combined with the integer programming algorithms described here to derive more robust algorithms.

REFERENCES

[1] Ballard, D. H. "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognition,* 13, 2, 1981.

[2] Ballard, D. H., and C. M. Brown *Computer Vision,* Englewood Cliffs, Prentice-Hall, 1982

[3] Bazaraa, M. S. and C. M. Shetty, *Nonlinear Programming,* John Wiley & Sons, New York, 1979.

[4] Davis, L. S., and A. Rosenfeld "Cooperating processes for low-level vision: a survey," TR-123, Dept. of Computer Sciences, University of Texas, Austin.

[5] Diamond, M. D., N. Narisimhamurthi, and S. Ganapathy "A systematic approach to continuous graph labeling with application to computer vision," Proc. AAAI Conf. on Art. Intell., Pittsburgh, July, 1 982.

[6] Diamond, M. D. "Report on an optimization approach to the edge linking problem In computer vision," *CRL-FN-1-83* Computer Research Laboratory, Department of Electrical and Computer Engineering, University of Michigan, Ann Arbor, Michigan, February, 1983

[7] Fu, K. S. Syntactic Methods In Pattern Recognition, New York, Academic Press, 1974.

[8] Freuder, E. C. "Synthesizing Constraint Expressions," Comm.ACM, 21,(11), 1978.

[9] Mackworth, A. K. "Consistency In Networks of Relations," Artificial Intelligence, 8, 1977

[10] Marr, D. "Analyzing natural images; a computational theory of texture vision." TR-334, AI Lab, MIT, June 1976.

[11] Montanari, U. "Networks of Constraints: Fundamental Properties and Applications to Picture Processing," Information Sciences, 7, 95-132 (1974).

[12] Perkins, W. A. "A model-based vision system for Industrial parts," IEEE Trans. Comp., Vol. C-27, No. 2, February 1978.

[13] Rosenfeld, A., R. A. Hummel, and S. W. Zucker "Scene labeling by relaxation operations," IEEE Trans. Syst., Man, Cybern., vol. SMC-6, pp. 420-433, 1976

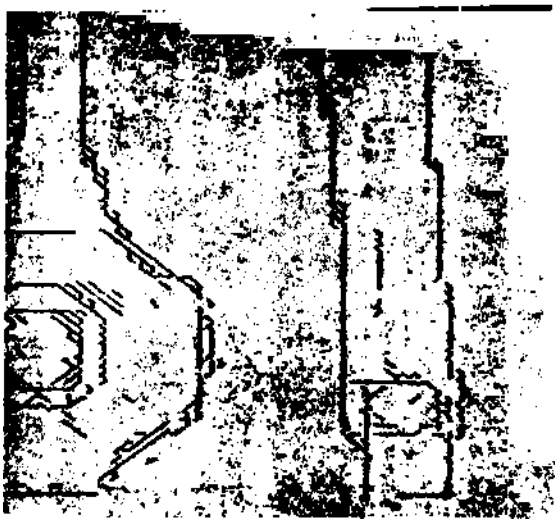
[14] Shapiro, J. F. Mathematical Programming; Structures and Algorithms, New York, John Wiley and Sons, 1979.

[16] Ullman, S. "Relaxation and constrained optimization by local processes," Comput. Graphics and Image Processing, vol. 10, pp. 115-125, 1979.

[16] Waltz, D. "Generating semantic descriptions from drawings of scenes with shadows," Ph.D. dissertation/ AI Lab, MIT, 1972.

[17] Zahn, C. T., and R. Z. Roskies, "Fourier descriptors for plane closed curves," IEEE Trans. Comput. C-21, 1972.

[18] Zucker, S. W., Y. G. Leclerc, and J. L. Mohammed, "continuous relaxation and local maxima selection: conditions for equivalence," TR 78-15R, Computer Vision and Graphics Laboratory, Dept. Electrical Engineering, McGill University, Montreal, Quebec, Canada, December, 1978.



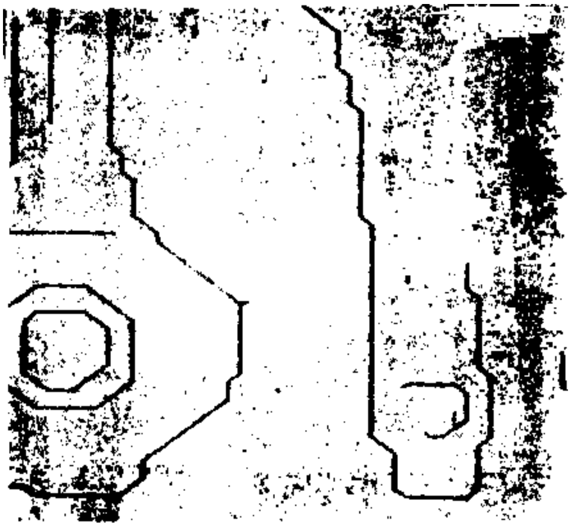


Figure 4a (above): initial labeling.

Figure 4b (above right): labeling after 8 iterations.

Figure 4c (right): labelinj after 20 iterations.